UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/879,838	06/12/2001	Ralph L. Beck	ATC-001 (5051/2)	3676

20210 7590 01/26/2005

DAVIS & BUJOLD, P.L.L.C.
FOURTH FLOOR
500 N. COMMERCIAL STREET
MANCHESTER, NH 03101-1151

EXAMINER	
RAMPURIA, SATISH	

ART UNIT	PAPER NUMBER
2124	

DATE MAILED: 01/26/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/879,838	BECK ET AL.	
	Examiner	Art Unit	
	Satish S. Rampuria	2124	

-- The MAILING DATE of this c mmunication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE *03* MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *05 October 2004*.

2a)☒ This action is **FINAL**. 2b)☐ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-14* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-14* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☐ The drawing(s) filed on _____ is/are: a)☐ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____ .
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

Response to Amendment

1. This action is in response to the amendment received on 12/01/2004.
2. The objection to trademarks is withdrawn in view of applicant's amendment.
3. The rejections under 35 U.S.C. 112 to claims 4, 7, 8, 9, and 10 are withdrawn in view of applicant's amendment.
4. Claims 1-5 and 7-14 are amended.
5. Claims 1-14 are pending.

Claim Rejections - 35 USC § 103

1. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

2. Claims 1, 8, 9, 10, 11, 12, and 13 are rejected under 35 U.S.C. 103(a) as being unpatentable over US Patent No. 6,314,558 to Angel et al., hereinafter called Angel, in view of US Patent No. 6,026,237 to Berry et al., hereinafter called Berry.

Per claim 1:

Angel disclose:

- A method for instrumenting object oriented virtual-machine-executable software comprised of class files containing bytecode including instructions, each class file being (col. 3, lines 16-17 "instrumenting a byte code computer program". also, see fig. 3), defined by a class (col. 4, line13-14 "storing an identifier for the object class") and each

class being associated with a class-hierarchy location and having a corresponding class name, (col. 7, lines 32-65 "The instrumentation software 63 uses the IR tree data element 66... operators and operands"), the method comprising the steps of:

- (a) identifying a class included in the software as a target class wherein the target class is (col. 5, line 66 "object code 46 is target-specific") associated with a first class-hierarchy location and with a first class name (col. 7, lines 32-33 "The instrumentation software 63 uses the IR tree data element 66");
- (b) instrumenting the target class by one of:
- (i) and assigning the new instrumented class to a class-hierarchy location adjacent to and above the first class-hierarchy location of the target class, whereby the instructions of the target class remain in an original un modified form (col. 6 and 7, lines 62-67 and 1 to 12 "compiler converts...to created the object code" col. 3, lines 22-23 "Instrumenting a portion of the byte code corresponding to a method call");
- (ii) and assigning the new instrumented class to a class-hierarchy location adjacent to and below the first class-hierarchy location of the target class, assigning the first class name to the new instrumented class and assigning a second class name to the target class whereby the instructions of the target class remain in an original un modified form(col. 6 and 7, lines 62-67 and 1 to 12 "compiler converts...to created the object code" col. 3, lines 22-23 "Instrumenting a portion of the byte code corresponding to a method call");
- (iii) creating an instrumented class of the target class by adding instrumentation to the target class without modifying the instructions within the target class (col. 6, lines 16-19 "Code instrumentation... object code 46") and retaining the first class name for the

new instructed class, whereby the target class becomes the instrumented class and the instructions of the target class remain in the, unmodified form; (col. 6 and 7, lines 62-67 and 1 to 12 "compiler converts...to created the object code" col. 3, lines 22-23 "Instrumenting a portion of the byte code corresponding to a method call"); and

- (c) causing a virtual machine to process the class having the first class name as the target class (col. 20, lines 62-63 "Byte code may be instrumented by instrumenting each class as the class is loaded by the VM runtime system").

Angel does not explicitly disclose (i) creating a new instrumented class separate from the target class, adding instrumentation to the new instrumented class; and (ii) creating a new instrumented class separate from the target class, adding instrumentation to the new instrumented class,

However, Berry discloses in an analogous computer system (i) creating a new instrumented class separate from the target class (col. 7, line 15 "The modified class file is then created (step 522)"), adding instrumentation to the new instrumented class (col. 5, line 21 "adding code for the purpose of instrumentation"); and (ii) creating a new instrumented class (col. 7, line 15 "The modified class file is then created (step 522)"), adding instrumentation to the new class adding instrumentation to the new class (col. 5, line 21 "adding code for the purpose of instrumentation").

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of creating a class and adding instrumenting to a class for as taught by Berry into the method of byte code instrumenting as taught by Angel.

The modification would be obvious because of one of ordinary skill in the art would be motivated to create new and add instrumenting to the class to provide code analyzer and performance measurement as suggested by Berry (col. 2, lines 7-31).

Per claims 8, 9, and 10:

The rejection of claim 1 is incorporated, and further, Angel does not explicitly disclose wherein the virtual-machine is a Java virtual machine and the steps of adding instrumentation to the target class include adding, removing, modifying, reordering or substituting a named class component of a set of named class components including: (i) a class name (ii) a super class name (iii) an interface index array (iv) a field table (v) a method table (vi) a constant pool (vii) an attribute table (viii) an index array, and (ix) access flags.

However, Berry discloses in an analogous computer system wherein the virtual-machine is a Java virtual machine and the steps of adding instrumentation to the target class include adding, removing, modifying, reordering or substituting a named class component of a set of named class components including (col. 5, lines 20-24 "Code may be added to, deleted from, or modified in the Java class file for many reasons, including instrumentation, benchmarking, performance tuning, modifying functionality, applying functional or performances patches" and col. 5, lines 29-33 "FIG. 5 illustrates the steps of transforming the class file components and then reconstructing the class file, specifically for the purpose of adding performance instrumentation code at the entry and exit of every method contained in the class file"): (i) a class name (ii) a super class name (iii) an interface index array (iv) a field table (v) a method

table (vi) a constant pool (vii) an attribute table (viii) an index array, and (ix) access flags. All of the components would be obvious in a class.

The feature of modifying class would be obvious for the reasons set forth in the rejection of claim 1.

Per claims 11 and 12:

The rejection of claim 1 is incorporated, and further, Angel does not explicitly disclose modifying the new instrumented class to recognize a super class associated with the target class as the super class associated with the new instrumented class; modifying the target class to recognize the new instrumented class as the super class associated with the target class.

However, Berry discloses in an analogous computer system modifying the new class to recognize a super class associated with the target class as the super class associated with the new instrumented class (col. 1, lines 60-62 "Object class have a superclass and that all field and method references in the constant pool have valid names, classes, and type descriptors"); modifying the target class to recognize the new instrumented class as the super class associated with the target class (col. 1, lines 60-62 "Object class have a superclass and that all field and method references in the constant pool have valid names, classes, and type descriptors").

The feature of modifying the class, as a super class would be obvious for the reasons set forth in the rejection of claim 1.

Claim 13 is the apparatus claim corresponding to method claim 1 and rejected under the same rational set forth in connection with the rejection of claim 1 above.

3. Claims 2, 3, 5, 6, 4, 7, and 14 are rejected under 35 U.S.C. 103(a) as being unpatentable over Angel, in view of US Patent No. 6,405,367 to Bryant et al., hereinafter called Bryant.

Per claims 2, 3, 5, and 6:

The rejection of claim 1 is incorporated, and further, Angel disclose:

- (aa) after completing step (a), operating a virtual machine to initiate loading and execution of the virtual machine executable software (col. 20, lines 62-63 "Byte code may be instrumented by instrumenting each class as the class is loaded by the VM runtime system").

Angel does not explicitly disclose (ab) after completing step (aa) suspending the operation of the virtual machine after loading and before linking the target class; (ca) after completing step (c), un-suspending operation of the virtual machine.

However, Bryant discloses in an analogous computer system (ab) after completing step (aa) suspending the operation of the virtual machine after loading and before linking the target class (col. 6, lines 56-57 "The application program 140 suspends processing until the return of data at step 145" also fig. 7); (ca) after completing step (c), un-suspending operation of the virtual machine (col. 6, lines 57-60 "After data is received from the server, the application program 140 unsuspends itself to receive the data and error output of the child Java server 180 and to receive any exit status at step 146" also fig. 7).

Therefore, it would have been obvious to a person of ordinary skill in the art at the time the invention was made to incorporate the method of suspend and un-suspending virtual machine as taught by Bryant into the method of byte code instrumenting as taught by Angel. The modification would be obvious because of one of ordinary skill in the art would be motivated to suspend and un-suspend the machine to provide the high performance execution of JAVA application as suggested by Bryant (col. 2, lines 10-24).

Per claims 4 and 7:

The rejection of claim 1 is incorporated, and further, Angel disclose:

- wherein the virtual-machine is a Java virtual machine (col. 20, lines 25-26 "virtual machine (VM) runtime system that interprets and runs byte code, such as Java byte code") and the step of identifying the target class included within the virtual machine executable (col. 20, lines 39-41 "the class instance 406 is provided as an input to the VM runtime module 404 which interprets and executes the executable steps of the class instance 406") software includes the steps of:
- (a) specifying a set of class attribute names and associated value descriptions matching class attribute names and associated values possessed by at least one class included in the virtual-machine-executable software, (col. 26, lines 21-26 "Processing at the step 612 may include modifying the native attribute of the method to convert the method to a byte code method, creating a new name for the native method and adding the new name as a private native method declaration, and adding byte code instructions to call the native method under the new name" and the rest of col. 26 and 27) the set including an

attribute name of a set of attribute names including (i) a class name (ii) an interface name (iii) a parent class name (iv) an inherited method name (v) a defined method name (vi) a private method name (vii) an inherited field name (viii) a defined field name (ix) a private field name (x) a constant value attribute (xi) a synthetic attribute (xii) a code attribute (xiii) an exception attribute, and (xiv) a depreciated attribute. All of the attributes would be obvious within a class.

- (c) classifying the class possessing a attribute name and associated value consistent with the set of specified class attribute names and associated value description as the target class. (col. 5, line 66 "object code 46 is target specific").

Angel does not explicitly disclose (b) searching for class possessing a class attribute names and an associated values consistent with the set of specified class attribute name and associated value description.

However, Bryant discloses in an analogous computer system (b) searching for class possessing a class attribute names and an associated values consistent with the set of specified class attribute name and associated value description (col. 7, lines 45-53 "The child Java server 180 then maps, at step 183, to the specified application (i.e., class and method) identified in the information that was communicated over the pipe connection and received at step 182... executes the specified application (i.e., class and method) using the specified program name, execution arguments").

The feature of searching for class and attributes would be obvious for the reasons set forth in the rejection of claim 2.

Claim 14 is the apparatus claim corresponding to method claim 4 and rejected under the same rational set forth in connection with the rejection of claim 4 above.

Response to Arguments

6. Applicant's arguments with respect to claims have been considered but they are not persuasive.

In the remarks, the applicant has argued that:

(i) For claim 1, applicant does not understand what point the examiner is making for the limitation (c) causing a virtual machine to process the class having the first class name as the target class.

(ii) Reference Angel does not teach, suggest or hint at fundamental and essential aspects of the present invention that are recited in claim 1 and 13.

(iii) Reference Berry does not teach or event suggest or imply either the creation of a new file containing the instrumentation or of inserting the instrumentation without modifying the instructions of the target class to be instructed. Applicant disagrees with examiner's interpretations of the teachings of Berry.

(iv) Applicant is disagree with examiner's interpretation of the relationship of the teachings of Berry regarding the limitation "superclass" as recited in claims 11 and 12.

(v) Applicant is disagree with examiner's interpretation as taught by Bryant of the operations of suspending the operation of a virtual machine between the loading and linking of a target class as recited in claims 2-7 and 14.

Examiner's response:

(i) Regarding the limitation as recited in claim, "virtual machine to process the class having the first class name as the target class". Examiner's has given a broad interpretation that virtual machine is processing a class file with any name given to it. Angel clearly teaches that instrumenting each by instrumenting each class as the class is loaded/processed by the virtual machine, which is very similar to the processing a class file by a virtual machine. Applicant only makes general allegations and does not point out any errors in the rejection. Therefore, the rejection is proper and maintained herein.

(ii) Reference Angel does teach instrumenting a bytecode of a computer program (see col. 3 and 4, lines 15-67 and 1-15). Angel discloses the limitations as recited in claim 1 and 13 (see rejection above). Applicant only makes general allegations and does not point out any errors in the rejection. Therefore, the rejection is proper and maintained herein.

(iii) As noted by Applicant that Berry does teach the modifications of the instructions of an existing file that is to be instrumented. Therefore, Berry does disclose reconstructing a class file for instrumenting purposes. Applicant only makes general

allegations and does not point out any errors in the rejection. Therefore, the rejection is proper and maintained herein.

(iv) Examiner interpretation regarding the "superclass", is as a superclass which is also known as parent class as described by the applicant. Rejecting under 103(a) and using the background part of the reference is appropriate. Applicant only makes general allegations and does not point out any errors in the rejection. Therefore, the rejection is proper and maintained herein.

(v) Bryant does teach the limitation recited in claims 2-7 and 14 regarding the suspending/un-suspending virtual machine. Bryant is an analogous art. Applicant only makes general allegations and does not point out any errors in the rejection. Therefore, the rejection is proper and maintained herein.

Conclusion

7. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event,

however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

Satish S. Rampuria
Patent Examiner
Art Unit 2124
01/24/2005

ANIL KHATRI
PRIMARY EXAMINER